# An energy-conserving collision-free MAC protocol for underwater sensor networks

Alfouzan, Faisal Abdulaziz; Shahrabi, Alireza; Ghoreyshi, Seyed Mohammad; Boutaleb, Tuleen

*Published in:*
IEEE Access

*DOI:*
10.1109/ACCESS.2019.2901646

*Publication date:*
2019

*Document Version*
Publisher's PDF, also known as Version of record

IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received February 8, 2019, accepted February 21, 2019, date of publication February 25, 2019, date of current version March 13, 2019.

*Digital Object Identifier 10.1109/ACCESS.2019.2901646*

# An Energy-Conserving Collision-Free MAC Protocol for Underwater Sensor Networks

**FAISAL ABDULAZIZ ALFOUZAN, (Member, IEEE), ALIREZA SHAHRABI, (Member, IEEE), SEYED MOHAMMAD GHOREYSHI, (Member, IEEE), AND TULEEN BOUTALEB, (Member, IEEE)**

School of Computing, Engineering and Built Environment, Glasgow Caledonian University, Glasgow G4 0BA, U.K.

Corresponding author: Faisal Abdulaziz Alfouzan (faisal.alfouzan@gcu.ac.uk)

**ABSTRACT** An underwater sensor network (UWSN) has recently attracted considerable attention due to its ability to discover and monitor the aquatic environment. However, its acoustic communication has posed several inherent characteristics, such as high latency, low available bandwidth, and high bit error rate. These unique characteristics have made contention-based medium access control (MAC) protocols inefficient for UWSNs. They are most expensive and are not as effective as they are in terrestrial networks. Through this principle, a contention-free MAC protocol is, therefore, considered to be more reliable and flexible to overcome the consequences of applying acoustic signals and also to achieve a high performance (improving the energy efficiency and throughput across the network) by eliminating the chance of collision. In this paper, we propose a novel energy-conserving and collision-free depth-based layering MAC (DL-MAC) protocol for UWSNs. DL-MAC is able to deal with the underwater MAC challenges, such as the near–far effect, spatial–temporal uncertainty, and hidden/exposed terminal problems. It is able to efficiently schedule the transmission and reception operations in each side by using the concept of layering and a distributed clustering algorithm. By using a TDMA-based principle, DL-MAC can assign separate time slots to every sensor node individually to access the medium without any possibility of collision. Our extensive simulation study shows that DL-MAC outperforms other protocols in terms of throughput, packet delivery ratio, energy consumption, and packets lost under varying traffic rates and the numbers of nodes.

**INDEX TERMS** Underwater sensor networks (UWSNs), medium access control (MAC), depth-based layering, distributed clustering approach, collision-free MAC protocol.

## I. INTRODUCTION

Underwater Sensor Networks (UWSNs) have recently been proposed as a powerful technology to explore and observe the ocean, which approximately covers two-thirds of the Earth's surface [1]–[3]. UWSNs are a class of ad-hoc networks distributed in underwater communication areas in which sensors communicate using acoustic signals. Compared with terrestrial networks in which radio signals are used, acoustic signals are subject to long propagation delay, low available bandwidth, and high dynamic channels. These inherent characteristics pose great challenges for underwater protocol design in UWSNs [4]–[7].

During the last decade, many studies have focused on the underwater environment and proposed various protocols for medium access control. These protocols use different scheduling techniques such as mobility-aware MAC protocols, centralized and distributed, and game theory-based methods [8]–[10]. The MAC protocols can generally be classified into two categories: contention-free and contention-based protocols [11]–[13]. For contention-free MAC protocols, communication channels are divided into time, frequency, or code domains such as Time Division Multiple Access (TDMA), Frequency Division Multiple Access (FDMA), and Code Division Multiple Access (CDMA) [14]. The second category, contention-based protocols, is further classified into MAC protocols either with or without Request-To-Send/Clear-To-Send (RTS/CTS) mechanism. The latter class, without RTS/CTS solutions, have been widely studied [15]–[17]. One of these protocol called Slotted-ALOHA [15] which degenerates to pure ALOHA with the long propagation delays that are characteristic of underwater aquatic environments. The performance of Slotted-ALOHA has then been improved by applying a guard time. In addition, two variants of ALOHA have been designed to achieve a better performance [17]. However, this class of solutions do not

The associate editor coordinating the review of this manuscript and approving it for publication was Mahdi Zareei.

perform well in multi-hop UWSNs specially when facing with hidden terminal problems.

Since the above approach cannot effectively prevent collisions and to provide valid solutions for multi-hop UWSNs, other MAC protocols with RTS/CTS mechanisms have been proposed. Floor Acquisition Multiple Access (FAMA) [18] one of those protocols, extends the transmission delays of RTS/CTS control packets to allow Multiple Access with Collision Avoidance (MACA) [19] operate in the networks with high latency such that in UWSNs. It still consumes more energy because of sending long control packets, however. Slotted Floor Acquisition Multiple Access (S-FAMA) [20] is another MAC protocol using RTS/CTS where the time is divided into a number of slots, and every packet can only be sent at the beginning of each slot. The RTS/CTS method is a one-way process which usually reserves the channel for only one transmission and reception process. Although energy efficiency is significantly improved in this method, the network throughput is commonly low due to the high delay over the handshaking process [3], [21].

UWSNs are expected to achieve a better performance with the above-mentioned MAC protocols. Some recent observations regarding the unique characteristics of UWSNs, such as long latency and a high bit error rate, mean that contention-based MAC protocols are very costly, however. Both classes, with RTS/CTS and without, cannot, therefore, perform as efficiently as expected [22]–[25].

Due to contentions-based MAC protocols being expensive to implement in UWSNs, a collision-free MAC is considered to significantly achieve a better performance. Besides, the long propagation delays in centralized MAC protocols typically take a long period of time to gather the global topology and transmission requests from all the sensors and then to inform them of the schedule, therefore a distributed solution is performed.

To minimize the control overhead, the network can be divided into different clusters [9]. The local data aggregation in a subset of sensors, as cluster heads, can lead to conserve more energy, and thus, contributes to increase the network lifetime [26]. Furthermore, the throughput can be increased by sharing slots across clusters while the remaining slots are available for future allocation. The distributed clustering approach can also increase the reliability and flexibility by controlling the packet transmission in local geographical areas [27]. Similar approaches of distributed clustering have recently been used in the literature such as Cluster-based Mobile Data Gathering scheme (CMDG) [28] and Graph Colouring MAC protocol (GC-MAC) [24].

In this paper, we present a Depth-based Layering MAC protocol (DL-MAC) which is able to schedule collision-free transmissions and receptions by considering the near-far effect, spatial-temporal uncertainty, and hidden/exposed node problems. Hence, DL-MAC can effectively provide energy-conserving by avoiding collisions and retransmissions using the network layering and cluster-based scheduling in each layer. In this way, separate time slots are assigned to each sensor node which contributes in more energy-conserving. In DL-MAC, an underwater communication area is equally divided into a number of layers, which are classified into three types. Every layer-type is assigned to a different frame. This is done to avoid the possibility of vertical collisions occurring between nodes located in adjacent layers. Layers with the same layer-type can thus operate in parallel, which are two layers' distance away from each other. To avoid any horizontal conflict that may occur between nodes in each layer, every frame is divided into a number of sub-frames. Every sub-frame is elected by cluster heads using a simple clustering approach. The cluster head collaborates with adjacent cluster heads to inform them about its own reserved sub-frame in a distributed manner. Every cluster head is eventually able to assign unique slots for itself and its cluster members to utilize during the operational window for data transmissions. Therefore, DL-MAC ensures collision-free transmissions and receptions of data packets by assigning a time slot to every node located in the same layer-type and sub-frame. It does not require any special techniques to eliminate collisions.

Furthermore, we obtain an upper-bound for the traffic rate which shows the maximum number of data packets can be transmitted during the network operation. This number can be changed based on the network topology and density. We also discuss the collision probability of the scheduling phase based on the scheduling interval and network density. This can help to decide about the duration of the scheduling phase in real-world scenarios.

The rest of the paper is organized as follows: Section II presents related works. Section III explains the MAC protocol challenges. Section IV introduces the DL-MAC protocol principles and design. Section V presents an upper-bound for the traffic rate in detail. Section VI discusses and analyses the probability of having collision among scheduling packet during the second phase. Section VII presents the performance of our proposed protocol against other MAC protocols through simulations. Finally, Section VIII draws the conclusions.

## II. RELATED WORK

In multi-hop wireless ad-hoc networks, it is essential to design MAC protocols to efficiently schedule sensors accessing the communication channel taking into consideration eliminating collisions and retransmissions as well as improving the energy efficiency and the network throughput. Due to the long propagation delay and narrow communication bandwidth in aquatic environments, the data packet transmissions consume much more energy than that in the terrestrial networks. These inherent features of underwater acoustic modems have significantly affected the performance of MAC protocols. While the underwater MAC protocols can typically be divided into two categories: contention-based and contention-free [11]–[13], [29], [30].

The first category, contention-based MAC protocols, is more reliable and manageable for dynamic network topologies; it is thus more suitable for underwater acoustic networks. These protocols can further be classified into two types: contention-based with an exchanging RTS/CTS scheme, and contention-based without RTS/CTS [25].

In the first class, contention-based with RTS/CTS, several MAC protocols have been proposed for UWSNs that have been adjusted from terrestrial networks [3], [20], [31]–[34]. A Distance-Aware Collision Avoidance Protocol (DACAP) [31] and S-FAMA [20], for instance, require exchanging information (i.e., RTS/CTS control packets) before sending data packets. Both protocols attempt to conserve energy by reducing collisions and retransmissions via handshaking and carrier sensing [21]. However, due to the unique characteristics of UWSNs, they achieve lower throughput and channel utilization than expected. Some approaches concentrate on scheduling transmissions simultaneously to achieve high network throughput and channel utilization. One of those approaches is Delay-aware Opportunistic Transmission Scheduling (DOTS) [3], which shows higher packet collisions because of failure overhearing, resulting in inefficient energy conservation.

MAC protocols in the second class (contention-based without a RTS/CTS scheme), usually let transmitter nodes send packets randomly after an initialization one-way contention, such as an adjusted ALOHA protocol for UWSNs [17], [35]–[37]. In [17], some extra control packets are added to avoid collisions. Some approaches utilize guard time slots to reduce packet conflicts [35], while [36] and [37] utilize a receiver-synchronized approach to minimize conflicts in slotted ALOHA. Nevertheless, they achieve low network throughput within heavy-traffic conditions. A Tone-Lohi (T-Lohi) is another approach proposed in [38], which utilizes a very small tone to reserve the contention round (CRs). Through this mechanism, T-Lohi is able to reduce the possibility of collisions. UWAN-MAC [16] is another contention-based MAC protocol without a RTS/CTS scheme, has also been proposed. Both protocols consume more energy in high offered loads because of more intensive channel competition and more hidden nodes, however. Furthermore, UWAN-MAC can only be applied in delay-tolerant UWSNs [11]. Contention-based MAC protocols, both with and without RTS/CTS schemes, are thus not as efficient as expected [22]–[24].

Because contention-based MAC protocols are mostly expensive in UWSNs, a contention-free MAC protocol usually achieves a better performance by employing either a scheduling-based or cluster approach to avoid collisions. Some cluster approaches and scheduling-based MAC protocols guarantee collision-free transmissions [8], [24], [25], [39]–[43]. A Graph Coloring MAC protocol (GC-MAC) [24] is one such protocol that inspired by graph coloring techniques to improve a reservation-based contention-free MAC protocol. By using a TDMA-based approach, GC-MAC is able to assign every time slot (color) to every particular sensor in the network in a distributed manner. Sensors that have the same colors can, therefore, send simultaneously with no chance of collisions to support spatial reuse (simultaneous transmitting in different neighborhoods). It requires, however, that the location of the reference points ($rp_s$), situated in the internal cube, is predetermined.

An Efficient Depth-based MAC protocol (ED-MAC) [39], [40] utilizes a duty cycle mechanism by allowing every sensor node in the network to assign a unique time slot based on its priority. It is thus considered a collision-free schedule when the concept of sub-slots is introduced into every slot to avoid collisions between two hidden nodes that are neighbors of another deeper node. As a result, ED-MAC highly improves the network performance. However, the number of slots is doubled in each round to avoid any possibility of collisions (i.e., a sensor is not aware of on-going transmission with its transmission that may result in a collision). Furthermore, it guarantees a collision-free schedule, however, it produces inefficient channel utilization. Besides that, ED-MAC is more efficient, with narrow communication networks rather than shallower underwater networks.

On TDMA-based MAC protocol is another scheduling-based collision-free approach proposed in [41], which combines an electromagnetic (EM) field with acoustic channel in a single platform. While, the EM requires a huge amount of power to propagate longer distances because its waves attenuate rapidly in underwater, it is only utilized to control the sensor node by performing very short packets. The acoustic channel is used to carry the data packets. Interference-free graph TDMA-based MAC protocol is a similar approach called (IG-TDMA) proposed in [42]. This MAC protocol can achieve the optimal network throughput, but it may not be feasible for high traffic networks because of its high computational complexity.

A Staggered TDMA Underwater MAC Protocol (STUMP) [43] is another typical collision-free MAC protocol that is similar to TDMA. In this approach, the scheduling of every node is fixed for the networks entire lifetime. This strategy considerably reduces the channel utilization if the nodes' traffic loads are significantly heterogeneous. A similar approach called Spatial-Temporal MAC protocol (ST-MAC) [25], which is also collision-free MAC protocol. It takes the advantages of using the global topology information by creating a conflict graph. Through this mechanism, ST-MAC is able to schedule all the sensor nodes with the conflict graph and to improve the performance of the network. However, it uses a centralized scheduling algorithm (i.e., it needs to collect the global network's topology information) which is costly to obtain in underwater acoustic communication because of long propagation delays and low transmission rates.

## III. MAC PROTOCOL CHALLENGES

Due to transmission mode consumes higher energy than receiving and idle modes, collisions and retransmissions should be prevented in order to improve the energy efficiency and the network throughput. Taking into consideration the unique characteristics of underwater acoustic channel, such as high propagation latency and low available bandwidth. These constraints significantly affect the design of MAC protocols due to the challenges described as follows:

### A. NEAR-FAR EFFECT

Due to the unique characteristics of underwater acoustic channels, the near-far effect is a major design challenge for MAC protocols [5], [7]. It is defined as being that when the received power for all nodes are not almost identical, signals from distant nodes cannot be received successfully. This requires that the transmission power of each node must be controlled. As is shown in Figure 1, the distance between $C$ and $A$ is significantly longer than the distance between $B$ and $A$. As a result, the receiver node $A$ receives different signal-to-noise ratio (SNR) levels of signals originating from each of the sender nodes, due to the high level of noise produced by sensor $B$'s signals.

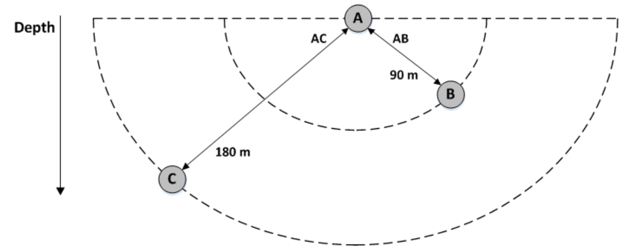


**FIGURE 1. Impact of the near-far effect on underwater MAC protocols.**

### B. SPATIAL-TEMPORAL UNCERTAINTY PROBLEM

Due to the inherent long propagation delay in acoustic communication environment, terrestrial medium access protocols are not applicable for UWSNs. Hence, it is important to consider the locations of the receiver sensors and the transmission times of the sender sensors to determine the status of the channel. This problem is called spatial-temporal uncertainty [25], [35], which can be defined as follows. Firstly, collision in the destination node (i.e., two packets arrive at the same time at the receiver node) depends on the transmission time of the sender node as well as the propagation delay. This case can be described as a relation between the location of the nodes and their transmission times. Secondly, the distance between nodes renders to uncertainty concerning the current channel status, and a collision may occur even if other nodes transmit their packet separately.

Due to the possibility that a high propagation delay could cause a collision, two examples of the spatial-temporal uncertainty problem are illustrated in Figure 2. Firstly, nodes $A$ and $C$ are able to transmit data packets concurrently, as is shown on the right hand side in this figure, because the reception time of the two data packets is classified when the propagation delays of the two senders are different. Another example is shown on the left hand side of the figure, illustrating that when both nodes $A$ and $C$ transmit data packets with different transmission times, a collision might occur at node $B$.

### C. HIDDEN AND EXPOSED TERMINAL PROBLEMS

The hidden terminal problem is defined as a terminal that is hidden inside the range of the intended destination of a packet, but outside the range of the transmitter. As depicted in Figure 3, the hidden node problem occurs when nodes $e$ and $u$ are visible from node $i$, but nodes $e$ and $u$ are invisible from each others. As a consequence, sending a packet from both nodes $e$ and $u$ may cause a collision in node $i$.

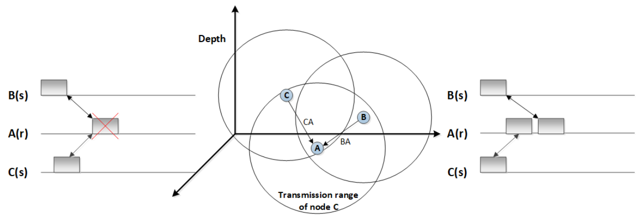


**FIGURE 2. Impact of the long propagation delay on underwater MAC protocols [40].**

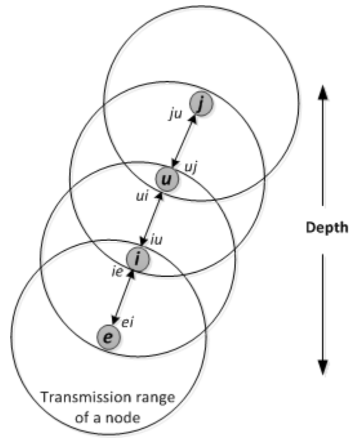


**FIGURE 3. Hidden and exposed terminal problems.**

The exposed terminal problem happens when nodes $i$ and $u$ are prevented from transmitting packets to their one-hop neighboring nodes $e$ and $j$ respectively. This is mainly because both sender nodes $i$ and $u$ are within each other transmission ranges, even though the receiver nodes $e$ and $j$ are out of each others transmission ranges, as shown in Figure 3. Specifically, if node $i$ transmits a packet to node $e$, node $u$ is prevented to transmit a packet to node $j$ after sensing the channel which might be interfered with the transmission by its one-hop neighbor node $i$. However, node $j$ still able to receive the transmission of node $u$ without interference.

## IV. DL-MAC PROTOCOL PRINCIPLES AND DESIGN

This section first presents the system assumptions which were considered in the DL-MAC protocol. It then introduces an overview of our proposed protocol before explaining how the network is modeled in the DL-MAC, followed by a detailed description of the DL-MAC design.

### A. SYSTEM ASSUMPTIONS

A multi-hop acoustic network is considered for sensor nodes to reach their respective destinations in a distributed manner using an omnidirectional and half-duplex acoustic modem.

In our underwater acoustic sensor network model, we assume that all sensor nodes are uniformly scattered in a three-dimensional (3D) underwater environment, and are represented by a graph, $G = (V, E)$, where $V$ is the set of vertexes (sensors), and $E$ denotes the set of communication links.

Figure 4 shows that a single sink is considered on the water surface, equipped with an acoustic modem for underwater communication with a radio modem for out-of-water communication with the monitoring center. Nodes those are located at the bottom of the water called (Anchored nodes) which set in predetermined locations to collect data. That data delivers to the sink through the relay nodes, those are located between a sink and anchored nodes at different layers (i.e., with different depths). These nodes not only forward the packets that have received from the anchored nodes or other neighboring nodes located in deeper layers, but also forward the packets that have generated themselves. Anchored and relay nodes utilize acoustic signals to transmit packets.

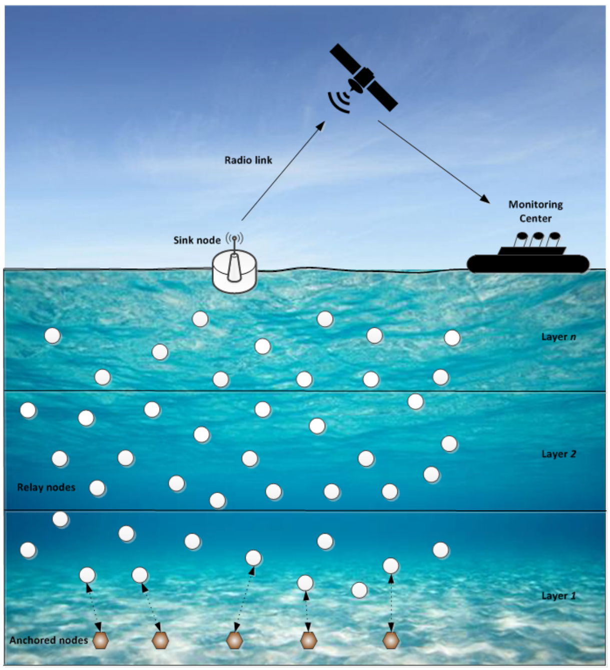


**FIGURE 4.** A multiple layer division architecture.

According to the sensors mobility in our underwater model, the relay nodes can move horizontally while their vertical movements are negligible because underwater sensors slightly move in the vertical direction [44], [45]. Sensors can also adjust their depth; however, the effect of depth adjustment is negligible because this procedure does not need to be performed more frequently. This model of mobility can meet the requirements of our DL-MAC protocol due to the nodes' depth which need to almost be constant during all the time.

### B. OVERVIEW OF DL-MAC PROTOCOL

To improve energy efficiency and reliability, the aquatic network area is divided into a number of horizontal layers, which are classified into three types. Each layer-type is assigned a different frame, as shown in Figure 6. In other words, every layer-type should be two layers away from each other. Layers with the same type can operate at the same frame (time). This is performed to avoid any possibility of vertical collisions. A distributed clustering approach for one-hop neighboring nodes is also used to eliminate any horizontal conflict between nodes in each layer-type. By using them, Depth-based Layering MAC (DL-MAC) protocol is able to resolve the near-far effect, spatial-temporal uncertainty, and any hidden/exposed terminal problems. By using a TDMA-based approach, DL-MAC is able to assign unique slots to every sensor node independently in the network in a distributed manner. Sensors, which are located two layers away from each other, can thus send at the same time with no chance of collision. Moreover, sensors located in adjacent layers or within the same layer are properly scheduled and therefore able to transmit or receive data packets without any conflict. DL-MAC trades off latency for high throughput, energy efficiency, and fairness, therefore, it is reliable and flexible to be used for different energy-critical applications in UWSNs.

DL-MAC includes three phases to operate, which are updating, scheduling, and operational phase, as illustrated in Figure 5. The beginning time of each phase has been set for all sensors in order to start and end together. A guard time has also been applied to avoid the effect of clock drift that may occur over a long period of time. In addition, a summary of the notations used to describe our algorithms is given in Table. 1.

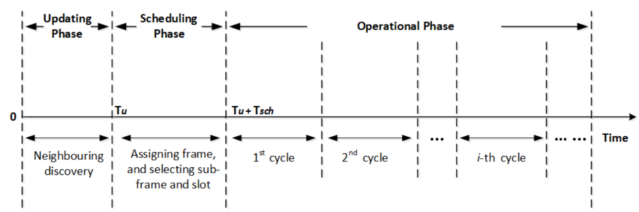


**FIGURE 5.** Timeline of DL-MAC protocol.

The primary goal of the updating phase is to collect information about one-hop neighboring sensors. This is performed by exchanging updating messages. The length of this phase, $T_u$, is set as a constant value for all sensors during the deployment time.

The purpose of the second phase, the scheduling phase, is to assign a unique slot to every individual sensor. By using a simple clustering approach, every cluster head is able to assign different time slots to all its cluster members, i.e., those that are located within a one-hop neighborhood. By the end of this phase, every sensor in the network has been assigned a different time slot in any nodes located in adjacent layers and clusters; hence, no collisions can occur. The length of the

**TABLE 1.** Notations used for explaining the DL-MAC algorithms.

| Terms | Definition |
|---|---|
| $T_u$ | Updating phase interval (s) |
| $T_{sch}$ | Scheduling phase interval (s) |
| $T_d$ | A degree timer (s) |
| $CH$ | A cluster head |
| $d_s$ | A sensor degree |
| $d_{max}$ | A maximum number of sensors in a one-hop neighbourhood graph |
| $\phi$ | A short random time duration (s) |
| $k$ | A number of sub-frames in each frame |
| $S_f - list$ | List of available sub-frames for each sensor |
| $S_f^{'} - list$ | A colouring list |
| $Slot_{set}$ | Available slots for each sub-frame |
| $M_s$ | A member slot |
| $msg$ | A cluster head message |
| $msg.h$ | The message hop count |
| $N_t$ | A neighbouring table of a sensor |

distributed scheduling phase, $T_{sch}$, is a predefined fixed value configured on each node prior to the deployment process. It should be long enough to allow nodes, from the seabed to the water surface, to reserve their slots but it is significantly shorter than that of the operational phase.

The operational phase is divided into a number of cycles, each consisting of three frames. Each frame is composed of $k$ sub-frames. Frames and sub-frames are used to avoid vertical layers and horizontal cluster interference respectively. Each sub-frame also encompasses a number of time slots. The number of time slots should be equal to the maximum number of nodes in a one-hop neighborhood graph (i.e., the maximum node degree, $d_{max}$, in a one-hop neighboring graph). At each cycle, from the information obtained from the scheduling phase, every sensor is aware of its frame, sub-frame, and its own reserved time slots, as well as the time slots reserved by its neighboring sensors. They can therefore be scheduled to wake-up either to transmit their own data packet during the reserved time slots, or possibly to receive a data packet from a neighboring sensor. They are asleep in the remaining time slots when there is no data transmission or reception. The length of this phase is also set to a predefined fixed value configured on each sensor before deployment.

### C. NETWORK MODEL

To increase the efficiency of the distributed MAC scheduling, the underwater network area is divided into a number of layers of equal size. Each layer width, $W$, is equal to the transmission range, $R_{tr}$, of a sensor node. The number of layers can then be calculated using $(Z_{area}/W)$, where $Z_{area}$ is the depth of the network area. This specification is used to avoid any transmission overlap between nodes located in adjacent layers.

Every sensor node exploits a pressure gauge embedded inside the node, to obtain its depth [39], [40], [46]. To determine which layer a sensor belongs to, $S_{lay}$, the depth information is used:

$$S_{lay} = (\frac{S_{depth}}{W})\ mod\ F_n, \qquad (1)$$

where $S_{depth}$ is a sensor depth in the network, and $F_n$ is the total number of layering types (layer-type A, B, and C) used. In each layer-type, sensors are also grouped using a clustering approach in order to avoid horizontal conflicts, as shown in Figure 6. As a result, sensors located in the same layer-type and cluster-group can operate concurrently without collision.

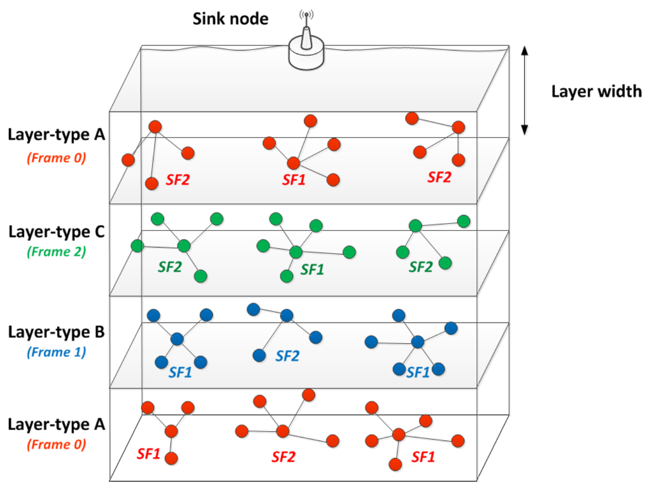


**FIGURE 6.** Horizontally layered structure of network topology.

### D. UPDATING PHASE

At the deployment process, the start time of this phase is set for all sensor nodes. During the updating phase, all nodes randomly broadcast a few small messages to discover their $d$-hop neighboring nodes. This message includes the ID and pressure of the sender. Upon receiving this message, each node immediately updates its neighboring table, $N_t$, based on the newly-discovered node. $d$ rounds of information exchanges are required to calculate the $d$-hop neighboring nodes of each sensor. To reduce the chance of collisions that may occur during this phase, sensors randomly set a transmitting timer for their updating message, afterwards, they set a new timer for the next updating message.

At the end of this phase, every sensor node is able to create its $d$-hop neighborhood graph. The length of this phase, $T_u$, is set to a predefined fixed value for all nodes. This should be long enough to allow nodes creating their $d$-hop neighborhood graph with accurate information. The length of this phase is very short compared to that of the operational phase, however. It should also be noted that for highly mobile scenarios, the total length of the updating, scheduling, and operational phases should be shortened to immediately react to topology changes in the network.

### E. SCHEDULING PHASE

To address any possibility of collisions that may occur vertically and horizontally, the concept of layering and a distributed clustering algorithm are utilized in our model. The primary goal of this phase is to schedule every individual node in the network to access the medium with no chance of collision.

- Addressing vertical conflicts can be achieved by dividing the network area into multiple layers, as was explained earlier. The layers are classified into three types; namely layer type A, B and C. Each layer-type is assigned a different frame, as shown in Figure 6. Layers of the same type are two layers away from each other in order to eliminate the chance of vertical collisions between nodes located in adjacent layers. Each layer-type has a separate transmission time to use during the operational phase. Layers of the same type share a frame and can thus operate concurrently.

- Addressing horizontal conflicts (i.e., any horizontal collision among nodes in each layer-type) can be conducted by dividing every frame into a number of sub-frames. Each sub-frame includes a number of time slots, a feature which is fully explained, in the following sub-section. A distributed clustering approach is used to allow the cluster heads, CHs, to select sub-frames for their clusters, which should be different from those of the adjacent clusters. Hence, clusters with the same sub-frame can transmit simultaneously without collisions. The CH is able to independently assign a unique slots for itself, and its cluster members to use during the operational phase for data transmission.

Using the information represented by the neighboring graph, every node knows its own $d$-hop neighboring nodes, defined by the number of nodes within its $d$-hop range. For the sake of simplicity and to avoid the overhearing of updating phase when $d > 1$, the current study assumes that $d = 1$.

In order to determine a CH, the proposed model gives higher priority to a node covering more nodes in its 1-hop range in the same layer. This feature can also yields a smaller number of CHs. Algorithm 1 details the scheduling phase. The basic idea is that each node competes with other nodes to become a CH based on its node degree. A node with the highest degree is distributedly elected as CH.

In the proposed algorithm, the node degree to elect a CH is applied via a timer-based approach. In this case, each node can set a degree timer upon starting the scheduling phase. The degree timer has an inversely proportional relation with the node degree. Nodes that have fewer neighbors within their neighborhood should be delayed for a longer period of time for their degree timers. The degree timer, $T_d$, for each sensor is given by:

$$T_d = (d_{max} - d_s) \times (T_{sch}/d_{max}) \pm \phi, \tag{2}$$

where $d_s$ denotes the sensor degree, and $d_{max}$ indicates the maximum node degree in the network topology. $d_{max}$ can be set at the deployment time based on the node density and network topology. $T_{sch}$ is the scheduling interval time, and $\phi$ is a short, random time duration used to differentiate the underwater sensor nodes with the same $d_s$.

During this period of time, each node listens to receive a CH message, *msg*, from the nodes with higher priority. Upon receiving the *msg*, every node within the neighborhood of the CH obtains its own slot. This message should be forwarded

**Algorithm 1** Collision-Free Scheduling Algorithm

**Procedure** Schedule Message
Set a degree timer $T_d = (d_{max} - d_s) \times (T_{sch}/d_{max}) \pm \phi$
*msg*: a new schedule message
**while** *the degree timer is not expired* **do**
  Listen and maintain the *msg*.
  **if** *(msg.layer == node.layer)* **then**
    **if** $(msg.h \leq d)$ **then**
      $node.CH \leftarrow msg.CH\_ID$
      $node.S_f \leftarrow msg.S_f$
      $node.slot \leftarrow myslot(msg.M_s)$
      **if** $(msg.h == d)$ **then**
        remove $CH\_ID$ & $M_s$ from *msg*
      $msg.h \leftarrow msg.h + 1$
      forwards the *msg*
    **if** $(d < msg.h \leq d + 3)$ **then**
      $msg.h \leftarrow msg.h + 1$
      update $node.S_f$*-list* based on $msg.S_f$
      forwards the *msg*
  **else**
    **if** $(msg.h \leq d)$ **then**
      remove $CH\_ID$ & $M_s$ from *msg*
      $msg.h \leftarrow msg.h + 1$
      forwards the *msg*
**if** *node has not received any CH msg* ***or*** *the degree timer is expired* **then**
  *node*.status $\leftarrow$ CH
  $msg.CH\_ID \leftarrow node.ID$
  $msg.S_f \leftarrow$ *First available* $S_f$ *from* $node.S_f$*-list*
  $msg.M_s \leftarrow \delta = \{(ID_i, Slot_i) \mid ID_i \in N_t, Slot_i \in Slot_{set}\}$
  Broadcasts *msg*
**End procedure**

up to $(d + 3)$ hops in order to avoid any conflict among adjacent clusters. In this way, each CH's message transmission does not collide with transmissions from other adjacent CHs, since each should have a different sub-frame for data transmission. Every one-hop neighboring node with CH therefore removes the CH's ID and member slot $(M_s)$ from the *msg* and broadcasts it. Thereafter, the message should be forwarded by each node if the message hop count is less than $(d + 3)$, in order to inform potential CHs.

Each CH reserves its own sub-frame by using the degree timer. In each frame (layer-type), every CH reserves a different sub-frame than its adjacent CHs, hence no adjacent CHs can obtain the same sub-frame. To guarantee no sub-frame is reserved by two adjacent CHs, a CH's message should be forwarded up to a $(d + 3)$ hop to inform adjacent CHs about its elected sub-frame. This is fully explained in Section VI.

After the degree timer expires, the sensor checks whether or not it has received any *msg* from a CH. If so, it has already been scheduled, and is a member of a cluster. Otherwise, the sensor node marks itself as a CH and then broadcasts its *msg*. This *msg* includes the CH's ID, sub-frame, and member slots.

At the end of the scheduling time interval, a non-CH sensor with no neighbors in its layer (i.e., no neighboring nodes located at the same layer) can select the first available sub-frame ($S_f$). Afterwards, it needs to broadcast a message, *msg*, to let its one-hop neighboring nodes, located in adjacent layers, to schedule their wake-up times to receive a data packet. This allows one-hop neighboring nodes to be in the listening mode to receive a data packet from a neighboring node located outside their layers. Finally, the one-hop cluster architecture implies that no two cluster heads can be within each other's neighborhood; i.e., they are at least two hops away from each other. This is achieved by delivering the CH's message up to $(d + 3)$ hops to inform the adjacent CHs about its schedule, hence no adjacent CHs can obtain the same sub-frame and neighbors. The length of this phase, $T_{sch}$, is set to a predefined fixed value.

To clarify some parts of algorithms, we present two examples that show how our algorithm can avoid vertical and horizontal collisions. In the first example, vertical example, Figure 7 illustrates how our proposed algorithm deal with two hidden CHs located in the same layer, which are both neighboring nodes of another node in a different layer.

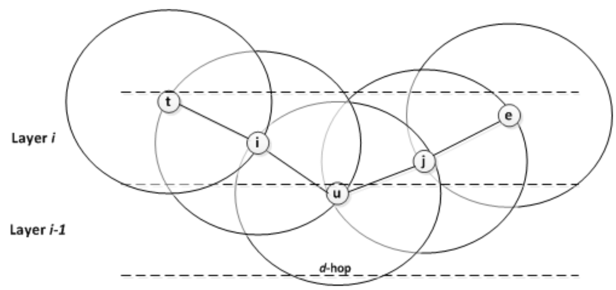


**FIGURE 7.** A specific vertical scenario may lead to a potential collision, which has been detected in DL-MAC.

In this figure, Layer $i$ is the adjacent upper layer of layer $i-1$ $(i > 1)$; while layer $i-1$ is the adjacent lower layer of layer $i$. By classifying these layers, layer $i$ can only be assigned a frame within a two-layer away from each others to prevent any possibility of collision. Based on this principle, our model allows a border node, located in adjacent layers, to participate with its own $d$-hop neighborhoods in adjacent layers during the scheduling phase. The purpose of letting the border nodes to interact with other neighboring nodes in adjacent layers is to avoid any possible collision and to ensure conflict-free operations between layers. This has been shown in Figure 7, when node $u$ interact with other neighboring CHs, nodes $i$ and $j$, which are hidden from each other and located in the same layer, none of them cannot elect the same sub-frame.

By assuming node $u$ does not participate with its 1-hop neighboring nodes located in adjacent layers, both nodes $i$ and $j$ reserve the same sub-frame because node $j$ is not aware of the sub-frame number reserved by node $i$, so it reserves the same sub-frame. As a result, sending a packet by sensors $i$ and $j$ may cause a collision in sensor $u$. This is the main reason of forwarding the CH's message up to $(d + 3)$ hops even between two adjacent layers to ensure collision-free transmissions and receptions for data packets.

Figure 8 is shown an example of how our model avoiding the horizontal collisions. In this figure, the sensor's ID and slot number are shown as a character and a number respectively. If a cluster (located in $R_3$) has two adjacent clusters (located in $R_1$ and $R_2$) in the same layer and both of them have sharing neighboring nodes (e.g., nodes $C$ and $B$ located between $R_1$ and $R_3$, and nodes $I$ and $H$ located between $R_2$ and $R_3$), all clusters ($R_1$, $R_2$, and $R_3$) should select different sub-frames. This can be done during the scheduling phase, when each CH transmits a message to $(d + 3)$ hops to notify the adjacent CHs about its elected sub-frame and hence avoiding any conflict may occur between them.

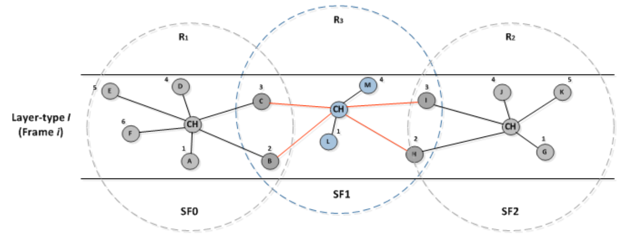


**FIGURE 8.** A specific horizontal scenario which may cause collisions. This has also been addressed in DL-MAC.

By assuming the clusters in $R_1$ and $R_2$ select a similar sub-frame, nodes $B$ and $H$ may reserve the same slot number 2. Likewise, nodes $C$ and $I$ also may reserve slot number 3. In this case, transmissions from sensor nodes within slot numbers 2 and 3 in both adjacent clusters simultaneously to the CH may therefore cause collisions, because conflict slot numbers have been reserved by nodes within the same sub-frame. This is another reason of transmitting a CH's message up to $(d+3)$ hops to avoid any conflict between adjacent CHs.

### F. OPERATIONAL PHASE

This phase is divided into a number of cycles, every cycle consisting of only three frames in order to eliminate vertical layer interference. Hence, layers with the same frame can simultaneously operate without collisions. Every frame also contains $k$ sub-frames to eliminate horizontal cluster interference. Therefore, clusters with the same sub-frame can concurrently operate according to their schedules. These sub-frames have already been reserved by the CHs during the second phase to avoid any conflict between adjacent clusters. The sub-frame is also divided into a number of slots. A unique slot is reserved for every cluster member during the scheduling phase. In this way, our proposed protocol ensures collision-free scheduling by assigning a different slot to each node in every two layers and clusters away from each others. Sensors with the same frame (layer-type), sub-frame, and slot can concurrently transmit data packets with no chance of collisions. The near-far effect, spatial-temporal uncertainty, and hidden/exposed node problems are thus fully addressed.

At each cycle, after obtaining the information from the scheduling phase, every sensor is aware of its sending time based on its own frame, sub-frame, and slot, as well as of the slots reserved by its neighborhood within the same frame and sub-frame. Hence, all sensors in the network can schedule to wake-up during their reserved time slots to send their own data packets or to receive a data packet from the neighborhood. They are in the sleeping mode during the remaining slots, if there is no data transmission and reception scheduled among them. This is repeated in each cycle. The length of the operational phase (i.e., the number of cycles) relies on the topology changes due to energy depletion or the displacement of nodes.

To calculate the length of each cycle, $T_{cy}$, it should be noted that $T_{cy}$ has a reverse relationship with the traffic rate, $\lambda$, which is presented in terms of packet per second. The higher the traffic rate, the shorter the cycle time, and hence also the shorter the sleeping time. The duration of each cycle time is given by:

$$T_{cy} = \frac{1}{\lambda}. \tag{3}$$

Every $T_{cy}$ is divided into three equal size frames. The length of frame, $T_f$, can be calculated:

$$T_f = \frac{T_{cy}}{3}. \tag{4}$$

Each frame consists of $k$ sub-frames, $S_f$. The length of sub-frame, $T_{S_f}$, can also be calculated:

$$T_{S_f} = \frac{T_f}{k}. \tag{5}$$

The sub-frame is also divided into a number of slots, $N_s$. Thus, the number of slots per sub-frame is proportional to the maximum node degree found in a $d$-hop neighborhood graph, which is given by:

$$N_s = d_{max}, \tag{6}$$

where $d_{max}$ denotes the maximum degree of sensors in a particular $d$-hop neighborhood graph, which is limited to the network topology. The length of each slot, $T_s$, can be given by:

$$T_s = \frac{T_{S_f}}{d_{max}}, \tag{7}$$

while:

$$T_s \geq T_{delay} + G_t. \tag{8}$$

More specifically, the length of each slot, $T_s$, should be equal to, or longer than, the maximum propagation delay along with a small guard time to ensure that a packet is entirely received at the destination before starting a data transmission by another node. $G_t$ indicates the guard time to avoid any possibility of collisions among the nodes, and $T_{delay}$ denotes the propagation delay, which can also be calculated using:

$$T_{delay} = \frac{R_{tr}}{V_s}, \tag{9}$$

where $R_{tr}$ is the transmission range of a sensor node, and $V_s$ is the velocity of sound in water.

## V. TRAFFIC RATE UPPER-BOUND ANALYSIS

The range of traffic rate, $\lambda$, which is used in Equation (3), should be defined for use in the proposed model. To find the upper-bound of the $\lambda$, the following equation must be satisfied:

$$T_s \leq T_{S_f}. \tag{10}$$

This means that the sub-frame length cannot be less than a slot length, which is already calculated in Equation (7). By replacing $T_{S_f}$ with Equation (5), it can be presented as:

$$T_s \leq \frac{T_f}{k}. \tag{11}$$

By replacing $T_f$ using Equation (4), it can be extended as:

$$T_s \leq \frac{T_{cy}}{3 \times k}. \tag{12}$$

$T_{cy}$ is further replaced with Equation (3), which can be represented as:

$$T_s \leq \frac{1}{\lambda \times 3k}. \tag{13}$$

Based on the above Equation, the upper-bound for $\lambda$ is calculated as:

$$\lambda \leq \frac{1}{T_s \times 3k}, \tag{14}$$

which shows that it depends on the slot length and the value of $k$. The slot length is a dynamic value which should be long enough to handle consecutively receiving packets, and $k$ is associated with the node deployment model and network topology.

## VI. SCHEDULING PHASE COLLISION ANALYSIS

This section discusses the collision probability of the scheduling phase. The scheduling interval has an impact on DL-MAC performance, and must therefore be precisely determined. In order to discuss the scheduling interval, for the sake of simplicity, we assume that each node chooses a random timer to become a CH.

In general, we need timers $x \in [a, b]$ with a probability density $f(x)$ such that [47]:

$$Prob[x_1 < X < x_2] = \int_{x_1}^{x_2} f(x) d_x, \tag{15}$$

with the corresponding distribution:

$$F(x) = Prop[X < x] = \int_a^x f(\acute{x})d\acute{x}. \quad (16)$$

A distribution of picking a random timer where all sensors have the same probability is called uniform distribution.

Every scheduling packet should be delivered up to $(d+3)$ hops before any of its $(d+3)$ hops neighbors send their scheduling packets. The delivery time of each scheduling packet in $(d+3)$ hops can be calculated as:

$$T = (d+3) \times (T_{delay} + P_{tx}), \quad (17)$$

where $T_{delay}$ is the maximum propagation delay which can be obtained by Equation (9), and $P_{tx}$ is a packet transmission duration, which can be given by $(L_{s_p}/B_{rate})$, where $L_{s_p}$ denotes the scheduling packet length and $B_{rate}$ indicates the bit rate of the acoustic modem. If we divide the scheduling interval $(T_{sch})$ by $T$, plus a guard time $(G_t)$, the number of timer selection choices can be calculated as:

$$C = \frac{T_{sch}}{T + G_t}. \quad (18)$$

If we assume that each sensor randomly broadcasts its packets within $T_{sch}$ interval, there is a probability of collision which depends on how many sensors are involved in a particular interval. Let us assume that only one sensor is participating, therefore, the probability of collision is 0. If there are $n$ sensors participating, the probability of collision for $n-th$ node is $(\frac{n-1}{C})$.

In general, the probability of collision during the given interval with $N$ sensors transmitting over $C$, the timer selection choices, is given as [48]:

$$P_{col} = 1 - \frac{(C-1)!}{C^{N-1}(C-N)!} = 1 - \prod_{i=1}^{N-1} \frac{C-i}{C}, \quad (19)$$

and the probability of no collision is:

$$P_{ncol} = \frac{(C-1)!}{C^{N-1}(C-N)!} = \prod_{i=1}^{N-1} \frac{C-i}{C}, \quad (20)$$

where $N$ denotes the average number of neighboring sensors in $(d+3)$ hops in each layer. By substituting from Equation (18), the probability of no collision among $N$ sensors during the scheduling interval can be written as:

$$P_{ncol} = \prod_{i=1}^{N-1} \frac{T_{sch} - i(T + G_t)}{T_{sch}}, \quad (21)$$

and the probability of collision is:

$$P_{col} = 1 - \prod_{i=1}^{N-1} \frac{T_{sch} - i(T + G_t)}{T_{sch}}. \quad (22)$$

If the scheduling interval increases, the probability of collisions decreases. In other words, the longer the $T_{sch}$, the higher the timer selection choices; hence, the lower the probability of collisions. More specifically, if the $T_{sch} \rightarrow \infty$, the probability of collision is Zero, as can be observed in the following equation:

$$\lim_{T_{sch} \rightarrow \infty} \left(1 - \frac{T_{sch} - i(T + G_t)}{T_{sch}}\right) = 0. \quad (23)$$

## VII. PERFORMANCE EVALUATION

This section first discusses the simulation settings of the proposed DL-MAC protocol. It then proceeds to evaluate important medium access metrics such as packet delivery ratio, throughput, energy consumption, and packets lost, before assessing the DL-MAC with different sub-frame configurations. We also compare DL-MAC with ED-MAC [39], [40], which is a distributed collision-free protocol, T-Lohi [38] and UWAN-MAC [16], which are both considered to be contention-based protocols.

### A. SIMULATION SETTINGS

This study implemented DL-MAC in Aqua-Sim, an NS-2 based simulator for underwater sensor networks [49]. Simulations were performed with the following parameters, unless otherwise noted. We randomly deployed the sensor nodes in a 3D narrow region of $300m \times 300m \times 600m$ for a fully connected network with an acoustic transmission range of $100m$. The bit rate for the acoustic modem is set to 10 kbps. The data packet length is 1000 bits, and all other control packets are set to 60 bits.

The power consumption on the transmission mode is 2 Watts, the power consumption on the receiver mode is 0.75 Watts, and the power consumption in sleep mode is 8 mW. In our simulation, we considered two parameters: traffic rate, and number of nodes. Regarding the first parameter, we randomly distributed 100 sensors into the given network. For the second parameter, all sensor nodes are deployed within the same area while increasing the number of nodes. In this case, the traffic rate is kept fixed to 0.1 packets per second. We considered $T_u$ as half a minute, and $T_{sch}$ as two minutes in our simulation settings. The range of the random time duration, $\phi$, is considered to be $\pm 10$ seconds during the degree timer, $T_d$. The simulation parameters are summarized in Table 2.

### B. PERFORMANCE METRICS

The performance metrics used in the simulation are throughput, packet delivery ratio, energy consumption, and packets lost as function of the traffic rate and number of nodes. These metrics are defined as follows:

$$\text{Throughput} = \frac{\text{Total packets successfully received} \times \Delta_{data}}{\text{Simulation time}}, \quad (24)$$

where $\Delta_{data}$ denotes the duration of transmitting a data packet, obtained by the corresponding propagation time plus the packet transmission time. The packet delivery ratio, PDR, is defined as the ratio of the packets successfully received in relation to the total packets generated in the network.

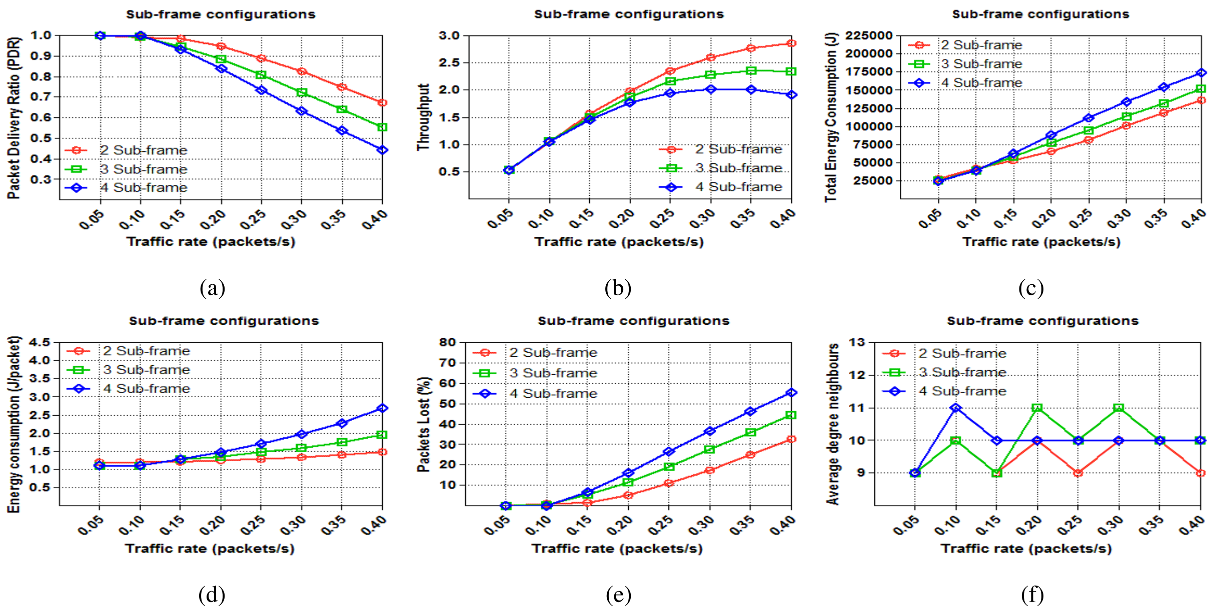


**FIGURE 9. PDR, Throughput, Total energy consumption, Energy consumption, Packets lost, and Average degree neighbors vs. Traffic rate. (a) PDR. (b) Throughput. (c) Total energy consumption. (d) Energy consumption. (e) Packets lost. (f) Ave. degree neighbors.**

**TABLE 2. Simulation parameters.**

| Parameter | Value |
|---|---|
| Transmission power | 2 Watts |
| Receiver power | 0.75 Watts |
| Idle power | 8 mW |
| Maximum transmission rage | 100 m |
| Bandwidth | 10 Kb/s |
| Acoustic propagation speed | 1500 m/s |
| Traffic rate | 0.05–0.40 packets/s |
| Node Number | 50–400 sensor nodes |
| Deployment region | 300 m × 300 m × 600 m |
| Movement model | RandomWalk 2D mobility model |
| Movement speed of nodes | 2 m/s, change movement direction every 2 s |
| Running rounds | 20 |
| Control packet size | 60 bits |
| Data packet size | 1000 bits |
| Length of the initial phase | 30 s |
| Length of the scheduling phase | 120 s |
| Simulation time of one round | 3600 s |

Energy consumption is obtained by dividing the overall energy usage in the network by the successfully delivered data packets, where the energy consumption is measured in joules per packet. The number of packets lost is defined as the total transmitted data packets that have not arrived successfully in relation to the amount of generated data packets.

### C. SIMULATION EVALUATION

In this section, the performance of DL-MAC is compared to that of ED-MAC, T-Lohi, and UWAN-MAC protocols through simulations. For each scenario, the results are averaged over 20 runs, each obtained with a randomly generated topology in each run. The total simulation time for each run is set to an hour. In these simulations, we first evaluate the DL-MAC with different sub-frame configurations under varying traffic rates and numbers of nodes. We then select the best configuration of DL-MAC among these parameters to be compared with other protocols in terms of packet delivery ratio, throughput, energy consumption, and packets lost under different sets of traffic rates and numbers of nodes.

#### 1) DL-MAC SUB-FRAME CONFIGURATIONS

The DL-MAC with three different sub-frame configurations is extensively studied under various traffic rates and numbers of nodes.

Figure 9 shows how the traffic rate affects the performance of DL-MAC with three sub-frame configurations by comparing the PDR, throughput, total energy consumption in joules, energy consumption joules per packet, packets lost, and average degree neighbors.

As can be seen in Figure (9a), the PDR of DL-MAC with two sub-frames delivers all the data packets successfully until the rate of 0.15 packets per second is reached; then a few packets dropped when the traffic rate further increases. DL-MAC with 2 sub-frame hence achieves a higher performance than other configurations, while other configurations are very efficient at low traffic rates. This is mainly because the traffic rate and the duration of the operational cycle have an inverse relationship; i.e., when the traffic rate further increases, the duration of the operational cycle decreases; hence insufficient number of slots to be occupied.

Figure (9b) shows the network throughput of DL-MAC across all sub-frame configurations. When the traffic rate increases, the network throughput correspondingly increases. In contrast, all DL-MAC sub-frame configurations achieve the same throughput with a low traffic rate up until a rate of 0.15 packets per second. As the traffic rate increases, the DL-MAC with 2 sub-frame configuration significantly increases, while other sub-frame configurations reach a saturation point.

Figure (9c) presents the total energy consumption in relation to increase the traffic rate. All DL-MAC configurations significantly consume a high level of energy when the traffic rate increases. We can observe that all DL-MAC configurations consume approximately the same amount of energy per joules at low traffic rates. When the traffic rate exceeds 0.15 packets per second, the DL-MAC with 2 sub-frames expends lower energy than other two configurations, followed by DL-MAC with 3 sub-frames, which consumes less energy than DL-MAC with a 4 sub-frames configuration by nearly 16%.

Figure (9d) shows the energy usage for each correctly received packet during the entire simulation. We can observe that DL-MAC with 4 sub-frames costs less energy per packet than other configurations with low data rates. However, it consumes more energy per packet than other configurations when the traffic load increases. The main reason for that is the DL-MAC's policy with more than 2 sub-frames is not applicable to an increase traffic rate, since the duration of the operational cycle has decreased. Consequently, the greater the sub-frame configurations, the fewer the number of slots, and hence a shortage arises of slots to be assigned.

Figure (9e) shows the percentage of packets lost as a relation with the traffic rate. When the traffic rate increases, the percentage of packets lost across all sub-frame configurations also increases. DL-MAC with 2 sub-frames performs better than DL-MAC with 3 sub-frames by almost 13%, and DL-MAC with 4 sub-frames by 25%. This is mainly because the operational cycle is divided into two sub-frames rather than 3 or 4 sub-frames, while the duration of the cycle deceases when the traffic load increases.

Figure (9f) shows the average degree neighborhood graph of the three sub-frame configurations of DL-MAC with the traffic rate. The range of degree nodes is between 9 and 11 nodes during low, medium, and high traffic rates. This is should be fluctuated from one another because the number of deployed nodes is fixed to 100 sensors while the node deployment process is performed randomly and uniformly.

Figure 10 illustrates how sparse and dense networks can affect DL-MAC's performance, as well as the scalability and flexibility among the sub-frame configurations of DL-MAC.

Figure (10a) shows that the packet delivery ratio of DL-MAC, with 2, 3, and 4 sub-frame configurations, is reduced to 28%, 23%, and 18% corresponding with 400 nodes, respectively. This is because a higher number of sub-frames under a fixed traffic rate implies the ability to handle more nodes.

As shown in Figure (10b), when the number of nodes increases, the network throughput of DL-MAC with all sub-frame configurations also significantly increases. In contrast, when the node density reaches 200 nodes and more, the throughput of DL-MAC with 4 sub-frames outperforms DL-MAC with 2 and 3 sub-frame configurations by almost 16% and 6% respectively. The reason for this is that a greater number of sub-frames in DL-MAC leads to more unoccupied slots to be assigned when the number of nodes increases.

The results shown in Figure (10c) illustrate that DL-MAC with 4 sub-frames achieves significantly superior energy

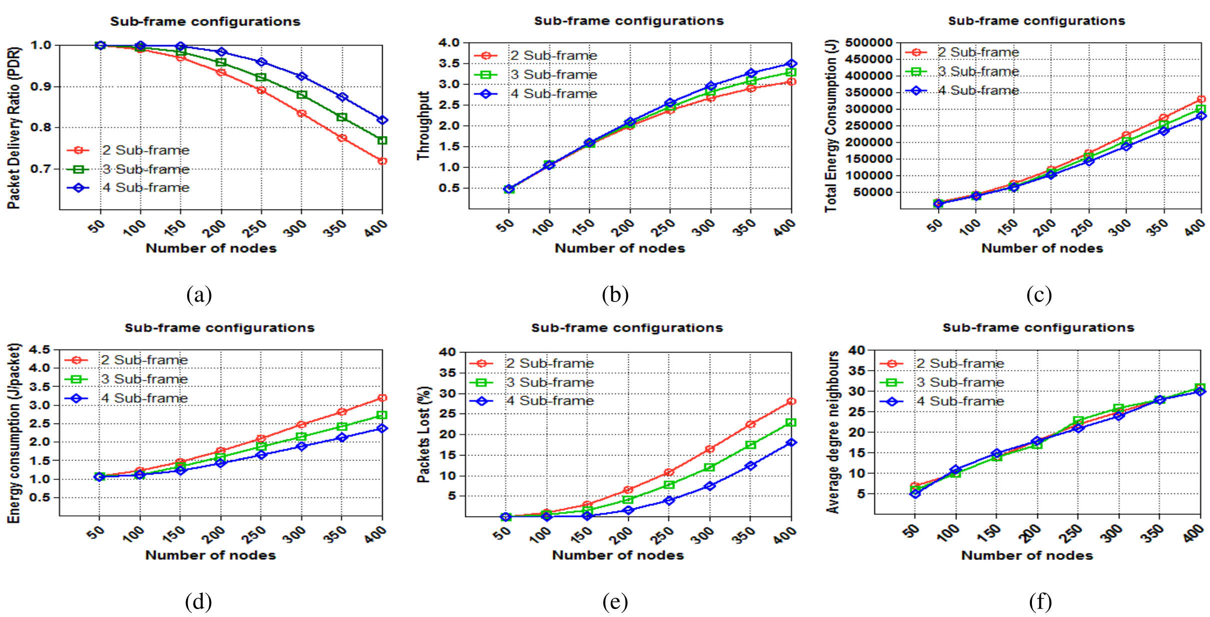


**FIGURE 10.** PDR, Throughput, Total energy consumption, Energy consumption, Packets lost, and Average degree neighbors vs. Number of nodes. (a) PDR. (b) Throughput. (c) Total energy consumption. (d) Energy consumption. (e) Packets lost. (f) Ave. degree neighbors.

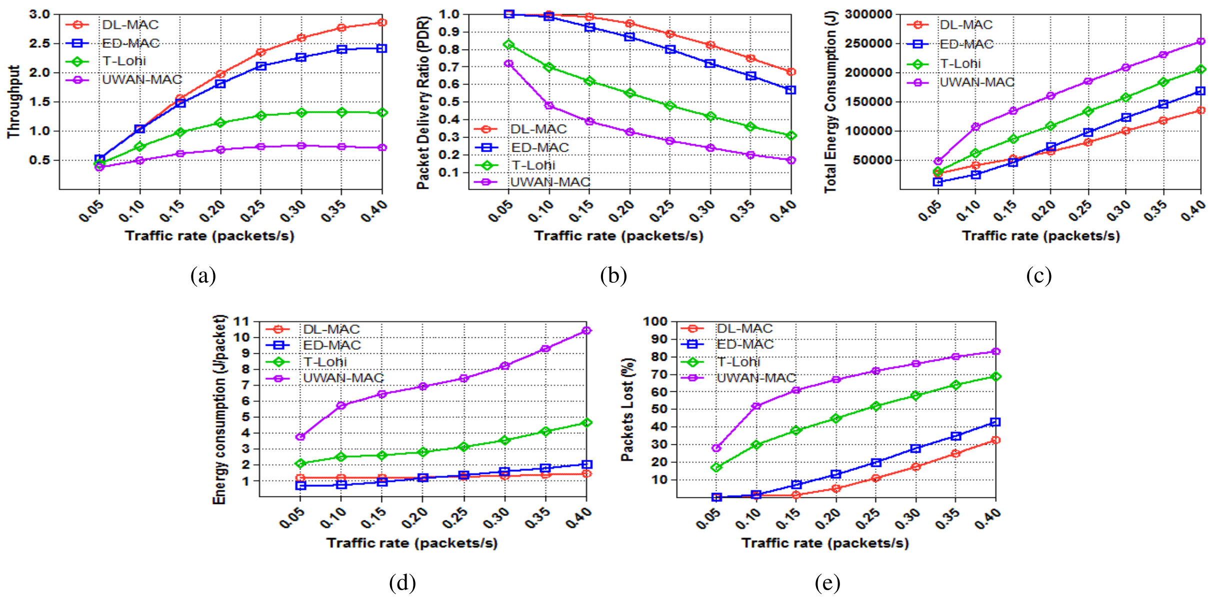


**FIGURE 11.** **Throughput, PDR, Total energy consumption, Energy consumption per received packet, and Packets lost vs. Traffic rates. (a) Throughput. (b) PDR. (c) Total energy consumption. (d) Energy consumption. (e) Packets lost.**

efficiency compared to DL-MAC with other sub-frame configurations. This is mainly because DL-MAC with 4 sub-frames delivers more packets than DL-MAC with other sub-frame configurations as the number of nodes increases. Hence, DL-MAC with a higher number of sub-frames is more scalable than DL-MAC with a lower number of sub-frames, as it is able to handle more data packets by allocating the extra slots to the nodes.

Figure (10d) shows the energy usage per packet of DL-MAC with the three sub-frame configurations as a function of the number of nodes. This figure shows that when the number of nodes increases, DL-MAC with 4 sun-frames consumes less energy per packet than DL-MAC with other sub-frame configurations. This is, as previously mentioned, because DL-MAC with more sub-frames allows more slots to be assigned by nodes when the number of nodes increases. It should be noted that increasing the number of nodes under a fixed traffic rate requires more slots to be differently assigned by any two-hop neighborhoods to avoid collisions.

Figure (10e) also shows the percentage of packets lost by DL-MAC with different sub-frame configurations. DL-MAC with 4 sub-frames is able to handle more packets than DL-MAC with other sub-frame configurations when the number of nodes increases, because DL-MAC with 4 sub-frames provides a higher number of slots than other sub-frame configurations of DL-MAC.

In Figure (10f), the average degree neighborhood graph of DL-MAC with all sub-frame configurations is shown as the number of nodes rises. As the number of nodes increases, the average degree neighboring nodes of each sensor node of DL-MAC with all sub-frame configurations also correspondingly increases, and eventually achieves a 30 degree neighborhood graph of each with 500 nodes. This is because the degree of nodes in a particular d-hop neighborhood graph depends on the node deployment model, and the network topology.

### 2) PERFORMANCE COMPARISON WITH OTHER PROTOCOLS

In the previous evaluation of DL-MAC with different $k$ sub-frame configurations, we select the best performance of DL-MAC with $k$ under varying traffic rates and numbers of nodes, equal to 2 and 4 sub-frame configurations respectively. Thereafter, the DL-MAC with these two configurations is compared with ED-MAC, T-Lohi, and UWAN-MAC in terms of throughput, packet delivery ratio, energy consumption, and packets lost under different sets of traffic rates and numbers of nodes.

Figure (11) shows how the load affects the performance of each protocol. Figure (11a) shows the network throughput of each of the four protocols at different traffic rates. As shown in this figure, the network throughput of the contention-free category (DL-MAC and ED-MAC) outperforms all other contention-based (T-Lohi and UWAN-MAC) protocols. As we expected, the benefits of the collision-free feature provide an optimal solution, as it is obtaining better throughput than the others. Although both DL-MAC and ED-MAC protocols utilise temporal and spatial reuse (concurrent sending in different neighborhoods), DL-MAC achieves a higher throughput than ED-MAC by almost 17%. This is because ED-MAC's scheduling policy does not take the horizontal two-hop neighborhood into account. In another way, the scheduling design of DL-MAC is superior to that of ED-MAC, as it considers all neighboring sensors vertically by grouping multi layers into three frames. Thereafter, each frame is divided into $k$ sub-frames to avoid any horizontal collision that may occur between the nodes. Hence, the DL-MAC protocol schedules all neighboring nodes vertically and horizontally, so that there is no chance of collision. In contrast, the number of slots depends on the maximum one-hop neighboring nodes in both protocols. Those time slots are doubled in ED-MAC to account for the hidden terminal nodes, while in DL-MAC those slots are exactly equal to the maximum one-hop neighborhoods. This means that hidden/exposed terminal nodes, as well as the near-far effect, are addressed using frames, sub-frames, and cluster approaches. Any possibility of collisions is thus eliminated.

It is expected that our proposed protocol performs better than T-Lohi because collision-free MAC protocols usually achieve a higher performance than contention-based approaches when the traffic rate is medium or high [25]. Compared to UWAN-MAC, DL-MAC is observed to perform approximately three times better with a medium traffic load, and four times better with a high traffic load. It is noteworthy that UWAN-MAC's performance is also impacted by the constraints posed by high latency, which lead to a low throughput [21].

As can be seen in Figure (11b), the PDR of DL-MAC performs well compared to other protocols under low and high traffic loads. ED-MAC, which is under the same classification as DL-MAC, delivers just under the DL-MAC by almost 17%. This is due to the fact that the scheduling of ED-MAC does not take the horizontal two-hop neighboring nodes into account, as DL-MAC does. However, when the traffic load gradually begins to increase, the PDR of T-Lohi and UWAN-MAC begin to rapidly decrease. This is because those protocols do not have the temporal and spatial reuse features as the DL-MAC and ED-MAC do.

Figure (11c) illustrates the total energy usage of all the protocols as a function of the traffic rate. DL-MAC and ED-MAC, on the one hand, as collision-free protocols, consume less energy than T-Lohi and UWAN-MAC, as contention-based protocols, in a narrow density topology. DL-MAC, on the other hand, consumes slightly more energy than ED-MAC with low traffic rates, due to the requirements of DL-MAC during the scheduling phase, such as the clustering approach and forwarding the scheduling message up to $(d + 3)$ hops to avoid any potential horizontal collision. Our proposed protocols can therefore improve network throughput by several orders of magnitude compared to UWAN-MAC and T-Lohi. This is mainly because UWAN-MAC does not detect hidden terminal problems, leading to more collisions and retransmissions. Even in T-Lohi, which is very efficient at low traffic rates, where contention rates are low, when the traffic rate increases, the energy cost increases marginally due to data collisions caused by incorrect reservations.

Figure (11d) shows the energy consumption per correctly delivered packet as a function of the traffic rate. We can first observe that DL-MAC expends less energy per packet than the other contention-based protocols, even though it achieves a higher throughput than all the other protocols during all traffic rates. We can also observe that UWAN-MAC consumes a greater energy per correctly received packet under all traffic rates. This is because UWAN-MAC involves unknown propagation delays, leading to more intensive competition to access the channel. Another reason for that is the latter protocol cannot avoid hidden terminal nodes, which also lead to increase the number of collisions and retransmissions.

The percentage of packets lost of all the MAC protocols as a function of the traffic rate is depicted in Figure (11e). When the traffic rate increases, the percentage of packets lost increases significantly, in contrast with DL-MAC protocol, which performs far better than all the other protocols under low and high traffic rates. This is mainly because DL-MAC considers the hidden/exposed nodes, the near-far effect, and the spatial-temporal uncertainty problems as well as employing a multi-layer division and a distributed clustering approach to avoid any potential collisions across the network.

Figure (12) shows how sparse and dense nodes can affect the performance of the DL-MAC, ED-MAC, T-Lohi, and UWAN-MAC protocols in order to study the network scalability and flexibility offered by each protocol. In this set of simulations, we keep the traffic rate fixed at 0.1 packets per second.

Changing the node density from 50 to 400 nodes generates the results displayed in Figure (12a). From this figure, an interesting phenomenon can be observed, which is that all protocols within 50 nodes can deliver almost the same amount of data packets. When the number of nodes increases, the throughput correspondingly also increases, and eventually achieves saturation point, except for the collision-free protocols (DL-MAC and ED-MAC) which continuously rise in throughput up to 3.5 and 2.8 packets per second with 400 nodes respectively. The throughput of the DL-MAC protocol outperforms all other protocols due to its specific benefits, such as highly scalable scheduling, which is able to handle more data packets than other contention-based protocols as well as solving any conflicts between nodes. We can also see that the throughput of T-Lohi reduces in a dense network, because there is more intensive competition between nodes to reserve the contention round (CR).

As shown in Figure (12b), the PDR of all the protocols is proportional to the number of nodes. The PDR decreases as the node density increases. This is because, in a network with lower density, the network is not saturated and there is enough number of available slots for the sensors. However, in a network with higher density, a higher number of sensors are assigned to a cluster head resulting in channel saturation and consequently some data packets are dropped in some clusters. However, the PDR of DL-MAC can handle more data packets when the node density increases. In contrast, DL-MAC successfully delivers all the data packets up to 200 nodes. But, when the node density is higher than 200, its PDR begins to slightly decrease. The reason for this is

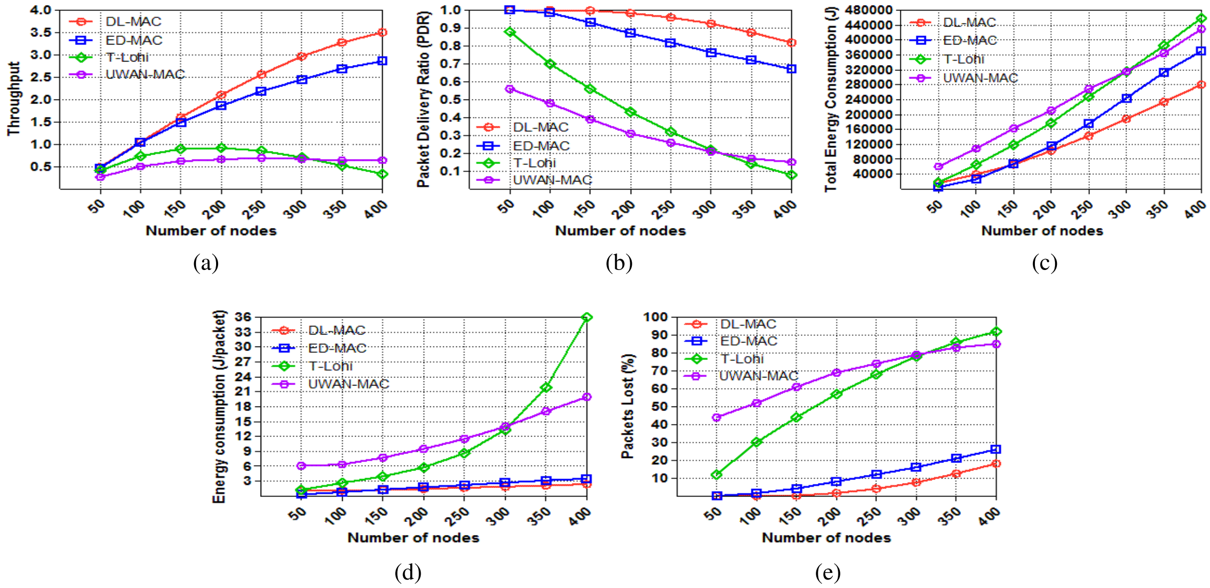


**FIGURE 12. Throughput, PDR, Total energy consumption, Energy consumption, and Packets lost vs. Number of nodes. (a) Throughput. (b) PDR. (c) Total energy consumption. (d) Energy consumption. (e) Packets lost.**

that it exceeds the channel threshold, resulting in the reduction of PDR. In contrast, the PDR of DL-MAC outperforms that of ED-MAC and other contention-based protocols with low and high density, mainly because of a number of sub-frame configurations that are implemented with DL-MAC which increase its ability to handle more nodes than others. However, T-Lohi's PDR rapidly decreases when the node density increases, due to an increasing number of tone packet collisions.

Figure (12c) shows that total energy consumption increases as the node density increases. Our proposed protocol consumes less energy than other contention-based protocols with low and high density, due to its features which allow it to eliminate any collision that may occur between nodes, both vertically and horizontally. However, UWAN-MAC expends a high amount of energy when the number of nodes increases because hidden terminal problems cannot be detected, which causes more collisions and retransmissions.

Figure (12d) shows the energy consumption per successfully delivered packet in relation to increasing node density. DL-MAC consumes the least energy across all the protocols, because it adopts energy conservation measures by considering the collision avoidance algorithm. This consideration makes collisions less likely, thus DL-MAC improves energy efficiency, throughput and fairness. This figure shows a huge difference in energy consumption between contention-free and contention-based protocols when the node density rises. This is due to the relationship between total energy consumption, as shown in the previous figure, and the data packets correctly delivered in the network. This relationship illustrates that the DL-MAC as a collision-free protocol consumes less energy and, at the same time, delivers a higher amount of data packets than contention-based protocols. This is can be proven with reference to the energy consumption of UWAN-MAC, which consumes much more energy than all the other protocols during all node density setups. This is caused by the inefficient scheduling of UWAN-MAC, which causes considerably more collisions and overhearing.

We can observe from Figure (12e) that the higher the number of nodes, the higher the challenges which are presented to reserving the channel. DL-MAC has the lowest percentage of packets lost during the increasing node density, because it is able to increase the efficiency of its scheduling by dividing the network area into multi layers, to avoid any potential vertical collisions between nodes located in adjacent layers, and then each layer includes $k$ sub-frames to be assigned by clusters which should be a two-way distance from each other, to eliminate any possibility of horizontal conflict. It should also be noted that the higher the $k$ sub-frame configurations under a fixed traffic rate, the greater the chance of handling more nodes.

## VIII. CONCLUSION

In this paper, we have proposed a novel distributed TDMA-based MAC scheduling protocol (DL-MAC) for underwater sensor networks. DL-MAC uses multiple layers and a clustering approach to eliminate the chance of collision, when accessing the media, by assigning different time slots to each individual node across the network. Through these techniques, DL-MAC is able to address hidden/exposed terminal problems, spatial-temporal uncertainty, and the near-far effect for UWSNs. It can consequently conserve more energy and improve the network lifetime. A time slot can be used by two nodes if, and only if, they are located in two-hop distances away from each other, both vertically and horizontally. We have also presented an upper-bound for the traffic rate. The probability of having collision among scheduling packet during the second phase has also been analyzed. Using an extensive simulation study, the performance of DL-MAC has been compared against those of other contention-based protocols recently reported in the literature. The results have illustrated the improvement achieved in terms of network throughput, PDR, energy consumption, and percentage of packets lost, with varying traffic rates and numbers of nodes.

For the future work, we plan to enhance the performance of the MAC protocol by using an adaptive scheduling phase, which considers a changeable traffic rate during the operational phase. In this way, all the setup process (i.e., updating and scheduling phases) is not required to be repeated. We also plan to use the mobile Autonomous Underwater Vehicle (AUV) in a distributed manner in order to design effective AUV employed data-gathering schemes for time-critical scenarios.

## REFERENCES

[1] F. Alfouzan, A. Shahrabi, S. M. Ghoreyshi, and T. Boutaleb, "Performance comparison of sender-based and receiver-based scheduling MAC protocols for underwater sensor networks," in *Proc. 19th Int. Conf. Netw.-Based Inf. Syst. (NBiS)*, Sep. 2016, pp. 99–106.

[2] R. W. L. Coutinho, A. Boukerche, L. F. M. Vieira, and A. A. F. Loureiro, "A novel void node recovery paradigm for long-term underwater sensor networks," *Ad Hoc Netw.*, vol. 34, pp. 144–156, Nov. 2015.

[3] Y. Noh, P. Wang, U. Lee, D. Torres, and M. Gerla, "DOTS: A propagation Delay-aware opportunistic MAC protocol for underwater sensor networks," in *Proc. 18th IEEE Int. Conf. Netw. Protocols*, Oct. 2010, pp. 183–192.

[4] E. M. Sozer, M. Stojanovic, and J. G. Proakis, "Underwater acoustic networks," *IEEE J. Ocean. Eng.*, vol. 25, no. 1, pp. 72–83, Jan. 2000.

[5] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: Research challenges," *Ad hoc Netw.*, vol. 3, no. 3, pp. 257–279, May 2005.

[6] J.-H. Cui, J. Kong, M. Gerla, and S. Zhou, "The challenges of building mobile underwater wireless networks for aquatic applications," *IEEE Netw.*, vol. 20, no. 3, pp. 12–18, May/Jun. 2006.

[7] J. Partan, J. Kurose, and B. N. Levine, "A survey of practical issues in underwater networks," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 11, no. 4, pp. 23–33, Oct. 2007.

[8] M. Zareei, A. K. M. M. Islam, C. Vargas-Rosales, N. Mansoor, S. Goudarzi, and M. H. Rehmani, "Mobility-aware medium access control protocols for wireless sensor networks: A survey," *J. Netw. Comput. Appl.*, vol. 104, pp. 21–37, Feb. 2018.

[9] A. Zeb *et al.*, "Clustering analysis in wireless sensor networks: The ambit of performance metrics and schemes taxonomy," *Int. J. Distrib. Sensor Netw.*, vol. 12, no. 7, Jul. 2016, Art. no. 4979142.

[10] D. Muhammed, M. H. Anisi, M. Zareei, C. Vargas-Rosales, and A. Khan, "Game theory-based cooperation for underwater acoustic sensor networks: Taxonomy, review, research challenges and directions," *Sensors*, vol. 18, no. 2, p. E425, Feb. 2018.

[11] Y. Han and Y. Fei, "DAP-MAC: A delay-aware probability-based MAC protocol for underwater acoustic sensor networks," *Ad Hoc Netw.*, vol. 48, pp. 80–92, Sep. 2016.

[12] C. Li, Y. Xu, B. Diao, Q. Wang, and Z. An, "DBR-MAC: A depth-based routing aware MAC protocol for data collection in underwater acoustic sensor networks," *IEEE Sensors J.*, vol. 16, no. 10, pp. 3904–3913, May 2016.

[13] S. Climent, A. Sanchez, J. V. Capella, N. Meratnia, and J. J. Serrano, "Underwater acoustic wireless sensor networks: Advances and future trends in physical, MAC and routing layers," *Sensors*, vol. 14, no. 1, pp. 795–833, Jan. 2014.

[14] D. Pompili, T. Melodia, and I. F. Akyildiz, "A CDMA-based medium access control for underWater acoustic sensor networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 4, pp. 1899–1909, Apr. 2009.

[15] L. F. M. Vieira, J. Kong, U. Lee, and M. Gerla, "Analysis of aloha protocols for underwater acoustic sensor networks," *Extended Abstract WUWNet*, vol. 6, Sep. 2006.

[16] M. K. Park and V. Rodoplu, "UWAN-MAC: An energy-efficient MAC protocol for underwater acoustic wireless sensor networks," *IEEE J. Ocean. Eng.*, vol. 32, no. 3, pp. 710–720, Jul. 2007.

[17] N. Chirdchoo, W.-S. Soh, and K. C. Chua, "Aloha-based MAC protocols with collision avoidance for underwater acoustic networks," in *Proc. 26th IEEE Int. Conf. Comput. Commun.*, May 2007, pp. 2271–2275.

[18] C. L. Fullmer and J. Garcia-Luna-Aceves, "Floor acquisition multiple access (FAMA) for packet-radio networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 25, no. 4, pp. 262–273, Oct. 1995.

[19] P. Karn, "MACA—A new channel access method for packet radio," in *Proc. ARRL/CRRL Amateur Radio 9th Comput. Netw. Conf.*, vol. 140, Sep. 1990, pp. 134–140.

[20] M. Molins and M. Stojanovic, "Slotted FAMA: A MAC protocol for underwater acoustic networks," in *Proc. OCEANS Asia–Pacific*, May 2007, pp. 1–7.

[21] P. Casari, B. Tomasi, and M. Zorzi, "A comparison between the Tone-Lohi and Slotted FAMA MAC protocols for underwater networks," in *Proc. OCEANS*, Sep. 2008, pp. 1–8.

[22] Y. Zhu, Z. Peng, J.-H. Cui, and H. Chen, "Toward practical MAC design for underwater acoustic networks," *IEEE Trans. Mobile Comput.*, vol. 14, no. 4, pp. 872–886, Apr. 2015.

[23] Y. Zhu, S. N. Le, Z. Peng, and J.-H. Cui, "DOS: Distributed on-demand scheduling for high performance mac in underwater acoustic networks," Dept. Comput. Sci. Eng., Univ. Connecticut, Storrs, CT, USA, Tech. Rep. UbiNet-TR13-07, 2013.

[24] F. Alfouzan, A. Shahrabi, S. M. Ghoreyshi, and T. Boutaleb, "Graph colouring MAC protocol for underwater sensor networks," in *Proc. IEEE 32nd Int. Conf. Adv. Inf. Netw. Appl. (AINA)*, May 2018, pp. 120–127.

[25] C.-C. Hsu, K.-F. Lai, C.-F. Chou, and K. C.-J. Lin, "ST-MAC: Spatial-temporal MAC scheduling for underwater sensor networks," in *Proc. IEEE INFOCOM*, Apr. 2009, pp. 1827–1835.

[26] S. M. Ghoreyshi, A. Shahrabi, T. Boutaleb, and M. Khalily, "Mobile data gathering with hop-constrained clustering in underwater sensor networks," *IEEE Access*, vol. 7, pp. 21118–21132, 2019.

[27] F. Alfouzan, A. Shahrabi, S. Ghoreyshi, and T. Boutaleb, "A comparative performance evaluation of distributed collision-free MAC protocols for underwater sensor networks," in *Proc. 8th Int. Conf. Sensor Netw. (SENSORNETS)*, Prague, Czech Republic, 2019, pp. 1–9.

[28] S. M. Ghoreyshi, A. Shahrabi, and T. Boutaleb, "A cluster-based mobile data-gathering scheme for underwater sensor networks," in *Proc. Int. Symp. Netw., Comput. Commun. (ISNCC)*, Jun. 2018, pp. 1–6.

[29] Y. M. Aval, Y. Han, A. Tu, S. Basagni, M. Stojanovic, and Y. Fei, "Testbed-based performance evaluation of handshake-free MAC protocols for underwater acoustic sensor networks," in *Proc. MTS/IEEE OCEANS*, Sep. 2016, pp. 1–7.

[30] S. Jiang, "State-of-the-art medium access control (MAC) protocols for underwater acoustic networks: A survey based on a MAC reference model," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 96–131, 1st Quart., 2018.

[31] B. Peleato and M. Stojanovic, "Distance aware collision avoidance protocol for ad-hoc underwater acoustic sensor networks," *IEEE Commun. Lett.*, vol. 11, no. 12, pp. 1025–1027, Dec. 2007.

[32] G. Xiaoxing, M. R. Frater, and M. J. Ryan, "An adaptive propagation-delay-tolerant MAC protocol for underwater acoustic sensor networks," in *Proc. OCEANS-Eur.*, Jun. 2007, pp. 1–5.

[33] Z. Peng, Y. Zhu, Z. Zhou, Z. Guo, and J.-H. Cui, "COPE-MAC: A contention-based medium access control protocol with parallel reservation for underwater acoustic networks," in *Proc. OCEANS IEEE Sydney*, May 2010, pp. 1–10.

[34] N. Chirdchoo, W.-S. Soh, and K. C. Chua, "RIPT: A receiver-initiated reservation-based protocol for underwater acoustic networks," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 9, pp. 1744–1753, Dec. 2008.

[35] A. A. Syed, W. Ye, J. Heidemann, and B. Krishnamachari, "Understanding spatio-temporal uncertainty in medium access with aloha protocols," in *Proc. 2nd Workshop Underwater Netw.*, Sep. 2007, pp. 41–48.

[36] Y. Zhou, K. Chen, J. He, and H. Guan, "Enhanced slotted aloha protocols for underwater sensor networks with large propagation delay," in *Proc. IEEE 73rd Veh. Technol. Conf. (VTC Spring)*, May 2011, pp. 1–5.

[37] P. Mandal, S. De, and S. S. Chakraborty, "A receiver synchronized slotted aloha for underwater wireless networks with imprecise propagation delay information," *Ad Hoc Netw.*, vol. 11, no. 4, pp. 1443–1455, Jun. 2013.

[38] A. A. Syed, W. Ye, and J. Heidemann, "T-Lohi: A new class of MAC protocols for underwater acoustic sensor networks," in *Proc. IEEE INFOCOM-27th Conf. Comput. Commun.*, Apr. 2008, pp. 231–235.

[39] F. Alfouzan, A. Shahrabi, S. M. Ghoreyshi, and T. Boutaleb, "Efficient depth-based scheduling MAC protoco for underwater sensor networks," in *Proc. 9th Int. Conf. Ubiquitous Future Netw. (ICUFN)*, Jul. 2017, pp. 827–832.

[40] F. Alfouzan, A. Shahrabi, S. M. Ghoreyshi, and T. Boutaleb, "An efficient scalable scheduling MAC protocol for underwater sensor networks," *Sensors*, vol. 18, no. 9, p. 2806, Aug. 2018.

[41] M. I. I. Alam and M. F. Hossain, "On TDMA based hybrid channel MAC protocol for underwater Sensor networks," in *Proc. 9th Int. Conf. Elect. Comput. Eng. (ICECE)*, Dec. 2016, pp. 574–577.

[42] R. Zhang, X. Cheng, X. Cheng, and L. Yang, "Interference-free graph based TDMA protocol for underwater acoustic sensor networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4008–4019, May 2018.

[43] K. Kredo, II, P. Djukic, and P. Mohapatra, "STUMP: Exploiting position diversity in the staggered TDMA underwater MAC protocol," in *Proc. IEEE INFOCOM*, Apr. 2009, pp. 2961–2965.

[44] S. M. Ghoreyshi, A. Shahrabi, and T. Boutaleb, "Void-handling techniques for routing protocols in underwater sensor networks: Survey and challenges," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 800–827, 2nd Quart., 2017.

[45] H. Yu, N. Yao, T. Wang, G. Li, Z. Gao, and G. Tan, "WDFAD-DBR: Weighting depth and forwarding area division DBR routing protocol for UASNs," *Ad Hoc Netw.*, vol. 37, pp. 256–282, Feb. 2016.

[46] Y. Noh, U. Lee, P. Wang, B. S. C. Choi, and M. Gerla, "VAPR: Void-aware pressure routing for underwater sensor networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 5, pp. 895–908, May 2013.

[47] M. Sugiyama, *Introduction to Statistical Machine Learning*. Burlington, MA, USA: Morgan Kaufmann, 2015.

[48] O. R. Popoola and S. Sinanović, "Design and analysis of collision reduction algorithms for led-based indoor positioning with simulation and experimental validation," *IEEE Access*, vol. 6, pp. 10754–10770, 2018.

[49] P. Xie *et al.*, "Aqua-Sim: An NS-2 based simulator for underwater sensor networks," in *Proc. OCEANS*, Oct. 2009, pp. 1–7.


**FAISAL ABDULAZIZ ALFOUZAN** received the B.Sc. degree in computer science and software engineering from the University of Hail, Hail, Saudi Arabia, in 2007, and the M.Sc. degree in network security from Glasgow Caledonian University, Glasgow, U.K., in 2014, where he is currently pursuing the Ph.D. degree. His current research interests include underwater sensor networks, designing the medium access control protocols, and distributed systems. He is a member of the IEEE Computer Society and VTS.

**ALIREZA SHAHRABI** received the B.Sc. and M.Sc. degrees in computer engineering from the Sharif University of Technology, Tehran, Iran, in 1991 and 1994, respectively, and the Ph.D. degree in computing science from the University of Glasgow, Glasgow, U.K., in 2003. Since 2001, he has been with the School of Engineering and Built Environment, Glasgow Caledonian University, where he is currently a Reader. His current research interests include network protocols, wireless, mobile and sensor networks, and also performance modeling and evaluation of parallel and distributed systems. He has published extensively in leading journals and well-established conferences. He has been on the editorial board of some international journals and also served the organizing and program committees of many international conferences and workshops. He is a member of the IEEE Computer Society.

**SEYED MOHAMMAD GHOREYSHI** received the B.E. degree from the Mazandaran University of Science and Technology, in 2009, the M.E. degree from the University of Tehran, in 2013, and the Ph.D. degree in computer networks and security from Glasgow Caledonian University, Glasgow, U.K. His current research interests include underwater sensor networks, designing the routing protocols, and void-handling techniques. He has served as a Reviewer for more than ten academic journals. He is a member of the IEEE Computer Society.

**TULEEN BOUTALEB** received the B.Eng. degree in electronics engineering and the Ph.D. degree in cellular mobile networks for telemetry from Glasgow Caledonian University, U.K., in 1995 and 2006, respectively. From 1995 to 1998, she was a Researcher on a European project working on the communications infrastructure. From 1998 to 2001, she was a Research Assistant working on a European project. Since 2001, she has been a Telecommunications Engineering Lecturer with the School of Engineering and Built Environment, Glasgow Caledonian University. Her research interests include cellular communication network performance enhancement, messaging in VANET, WSN, and satellite communications for remote monitoring.